

1/5

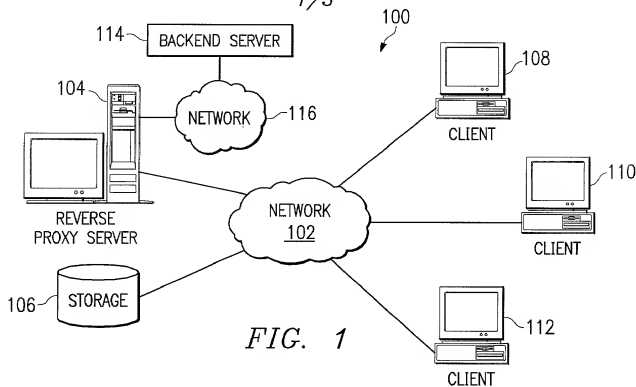


FIG. 1

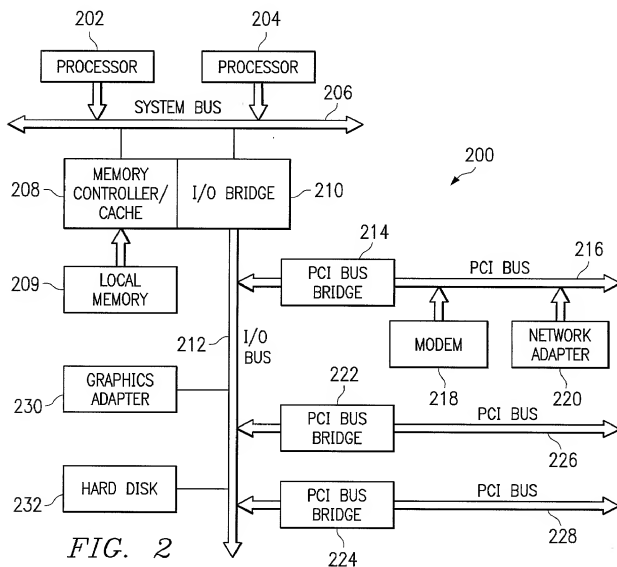


FIG. 2

2/5

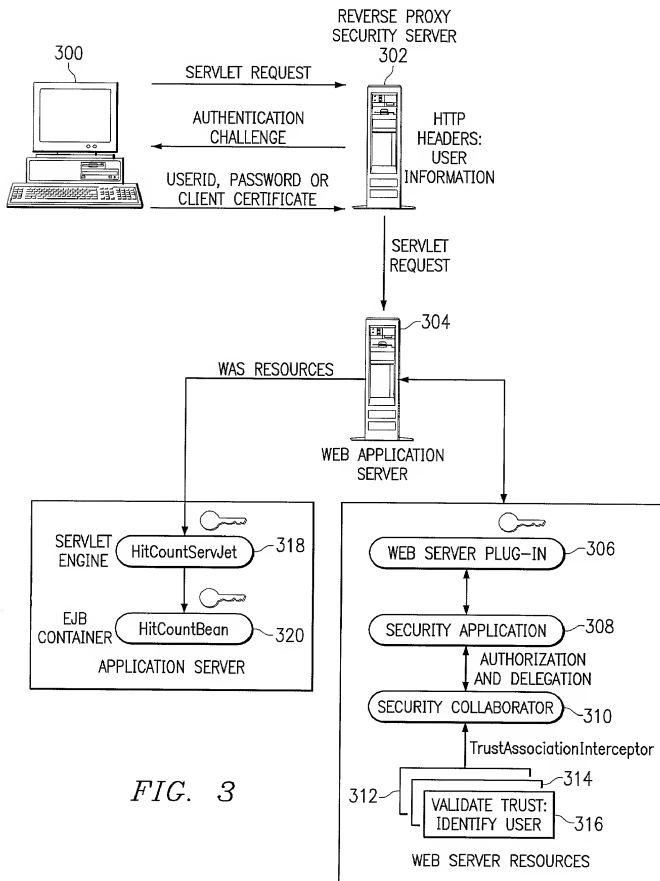


FIG. 3

400

FIG. 4A

com.ibm.websphere.security.trustassociation.enabled=true 402
 com.ibm.websphere.security.trustassociation.types=webseal36 404
 com.ibm.websphere.security.trustassociation.webseal36.interceptor=com.ibm.ejs.security.web.
 WebSealTrustAssociationInterceptor
 com.ibm.websphere.security.trustassociation.webseal36.config=webseal36a 408 406

410

FIG. 4B

webseal36a.properties: 412
 com.ibm.websphere.security.webseal36.id=iv-creds
 com.ibm.websphere.security.webseal36.hostnames=vivaldi.raleigh.ibm.com, vivaldi
 com.ibm.websphere.security.webseal36.ports=444

FIG. 5

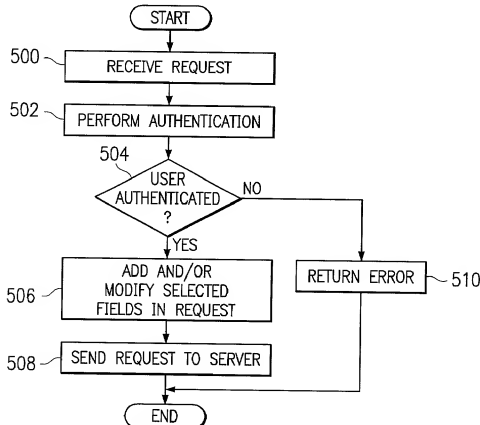
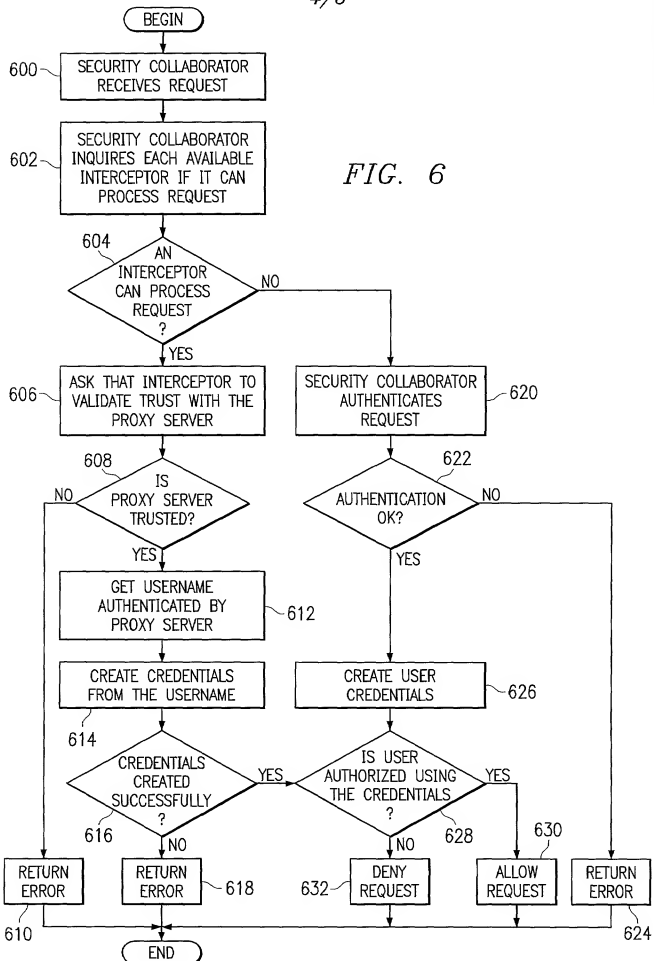


FIG. 6



5/5

700

```
package com.ibm.websphere.security.web;
public interface TrustAssociationInterceptor
{

```

```
/**
Every interceptor should know which HTTP requests originate from
the third party server that it is supposed to work with.
Given an HTTP request, this method must be used to determine whether
or not this interceptor is designed to process the request, in
behalf of the trusted server it is designed to interoperate with.
If the return value is false or an exception is thrown, then
WebSphere will consider that the request is not routed via the
trusted proxy server the interceptor is designed to handle.
WebSphere will pass the request to the next interceptor till there
is no more interceptors available, in which case it will be treated
to be a directly submitted request.

```

```
*/
public boolean isTargetInterceptor(HttpServletRequest req)
throws WebTrustAssociationException;
/**

```

This method is used to determine whether the interceptor trusts the server through which the request has been routed. This may involve authenticating the server in some manner. All the required Information to perform this operation should be available in the HTTP request.

If the third party server failed the validation, or is unable to provide the required information, a WebTrustAssociationFailedException must be thrown. This would be treated as an authentication failure and WebSphere would deny access to the requested secure resource.

```
*/
public void validateEstablishedTrust (HttpServletRequest req)
throws WebTrustAssociationFailedException;
/**

```

This method is used to retrieve the username of the end client (or the originator of the HTTP request). This method will be invoked if the validateEstablishTrust method invocation was successful.

The method returns a string. A return value of null or a WebTrustAssociationUserException should be thrown if the username is not available in the request header or the implementation determines that the username provided was invalid (based on some criteria, e.g., a list of valid usernames may have been decided earlier).

```
*/
public String getAuthenticatedUsername (HttpServletRequest req)
throws WebTrustAssociationUserException;

```

FIG. 7